UNIVERSITÉ D'AIX-MARSEILLE, MEDITERRANEAN INSTITUTE OF OCEANOGRAPHY (MIO);
UNIVERSITÉ DU SUD TOULON-VAR; CNRS-INSU/IRD UM 110; FRANCE.

Étude et modélisation des tourbillons océaniques avec le logiciel WATERS

Tuteur de stage : Mr Andrea M. DOGLIOLI

Jeanne CASTILLE

19/06/2014

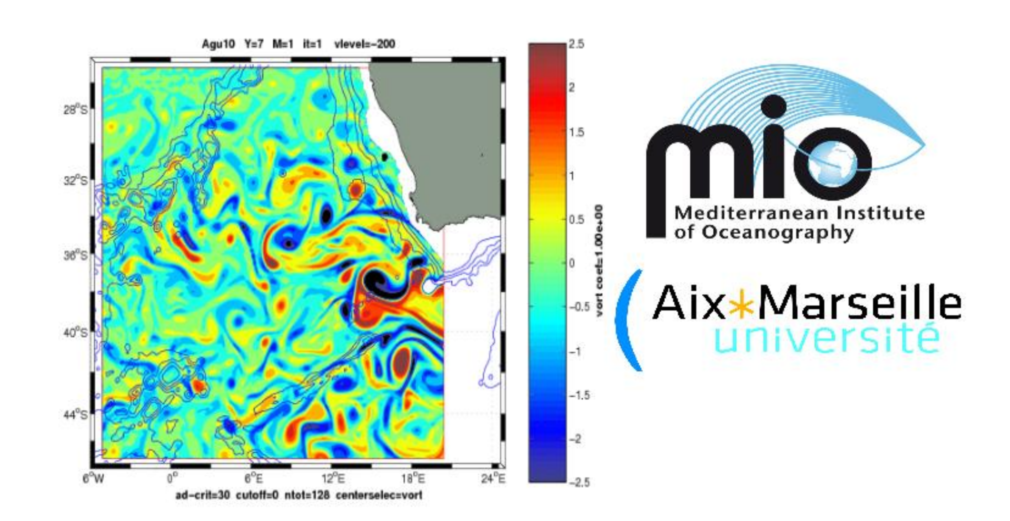


Table des matières

Résumé

Depuis 40 ans, les tourbillons océaniques ont été le sujet de nombreuses études expérimentales et théoriques. Ils peuvent avoir une profonde influence sur la productivité biologique, l'écologie et la biogéochimie océanique, et donc sur les cycles et flux élémentaires régissant l'océan. Plusieurs méthodes ont été proposées pour l'identification et le suivi de ces structures au sein de la circulation océanique : celles se basant sur les propriétés physiques du flux et les autres basées sur les critères géométriques. Les chercheurs du MIO ont donc développé un programme basé sur des critères physiques permettant à l'utilisateur d'identifier des structures tourbillonnaires en trois dimensions, ainsi que de suivre leur évolution et leur déplacement. Le package *WATERStools_0.0*, créé pour MATLAB (matrix laboratory) et qui regroupe tous les fichiers nécessaires au suivi et à la modélisation des tourbillons océaniques, fut conçu en 2007. Cette version étant maintenant devenue obsolète pour les nouveaux systèmes d'exploitation, plusieurs modifications ont été apportées afin de la rendre utilisable par tous. De plus, de nouvelles fonctionnalités ont aussi été ajoutées pour une interface plus compréhensible et intuitive. Toutes les étapes de la mise à jour sont décrites dans ce compte-rendu.

Abstract

Oceanic eddies have been the subject of many experimental and theoretical studies in the last 40 years. They can have a strong influence on biological productivity, upper ocean ecology and biogeochemistry, and thus in element cycles and fluxes. Several methods have been proposed for the identification and tracking of these structures in the ocean circulation: ones based on physical properties of the flow, and others based in geometric criteria. MIO researchers have developed a program based on physical criteria which allows users to identify three-dimensional eddy structures, and to follow their evolution and displacement. The *WATERStools_0.0* package, designed for the tracking and modeling of oceanic eddies in MATLAB, has been created in 2007. Since this version become obsolete for new OS, some modifications has been introduced to make it more user friendly. Besides, new features have been added for a more understandable and intuitive interface. Each update step is described in this report.

Introduction

Les tourbillons sont omniprésents dans le monde océanique et jouent un rôle majeur dans la dynamique des océans comme dans les transports de chaleur et de masse. Ce sont des vortex marins parfois étendus sur des centaines de kilomètres, et pouvant durer pendant des mois. Plusieurs méthodes de suivi de ces structures ont été proposées au fil des ans, certaines se basant sur les critères géométriques et d'autres sur les caractéristiques physiques des tourbillons.

L'utilisation des critères géométriques est moins fréquente que celle des paramètres physiques, une étude récente a réussi à identifier plusieurs vortex (Penven et al., 2005; Moolani et al., 2006; Chaigneau et al., 2007; Henson and Thomas, 2008). Cette méthode suppose que les tourbillons océaniques sont assimilables à un modèle de flux quasi-circulaire. La pente des lignes de courants instantanées peuvent donc être utilisées pour l'identification de structures. Deux techniques principales peuvent être appliquées à cette méthode : la technique de "centre de courbure" (Leeuw and Post, 1995) et la technique "d'enroulement de l'angle" (Sadarjoen and Post, 2000). Cependant, cette seconde technique montre de meilleurs résultats que la première.

Les méthodes basées sur les caractéristiques physiques identifient les tourbillons en utilisant des valeurs d'un paramètre spécifique qui dépasse un seuil choisi. Par exemple, les études menées par McWilliams (1990), représentant l'un des premiers travaux dans la détection automatique de structure tourbillonnaire, sont basées sur la notion que la rotation est le paramètre dominant d'un vortex. La vorticité relative ξ fut le paramètre physique utilisé pour la détection. Dans la catégorie des algorithmes basés sur des critères physiques, l'un des plus largement utilisé est basé sur le paramètre d'Okubo-Weiss, 'W'. Ce paramètre est calculé à partir du champ horizontal de vélocité comme suit :

$$W = s^2_{sh} + s^2_{st} - \xi^2$$

où s_{sh} et s_{st} représentent les déformations de pression et de cisaillement (en anglais strain et shear) et ξ la composante verticale de vorticité.

Une autre méthode est celle proposée par Doglioli et al. (2007), se basant sur les travaux de Siegel et Weiss (1997), et c'est à cette dernière que nous nous intéresserons dans ce compte-rendu. Comme dans la méthode proposée par McWilliams (1990), le paramètre physique utilisé pour la détection de tourbillon est ξ. La vorticité relative ξ est décomposée par analyse en ondelettes en deux dimensions et étendue dans la base qui minimise le coût de la fonction (entropie de Shannon). Ensuite, seules les ondelettes ayant le plus grand facteur sont utilisées pour reconstruire un champ 'lissé' pour lequel les tourbillons sont identifiés comme étant des régions connectées où $\xi \neq 0$ et où le centre de chaque tourbillon est défini comme un point du maximum local de $|\xi|$. Parfois, des filaments de petite échelle

sont détectés et analysés en tant que vortex. Afin de minimiser cette erreur, ces filaments sont automatiquement exclus de l'analyse en imposant une contrainte sur la largeur minimale des structures cohérentes détectées.

Objectifs du stage

Le but du stage était de tester la version du code WATERS qui n'était ni fonctionnelle ni aboutie, en plus de comprendre les principes à la base de la modélisation et du traçage des tourbillons océaniques. Après avoir effectué ce travail il était nécessaire de développer un programme de démonstration et écrire un mode d'emploi en anglais (Userguide) afin que n'importe quel internaute puisse utiliser le logiciel simplement.

Matériel et méthode

Comme il s'agit d'un logiciel informatique développé pour MATLAB, tout le travail réalisé durant le stage a été effectué sur ordinateur. Dans un premier temps en travaillant sur le logiciel MATLAB, et dans un second temps sur le logiciel LaTeX.

Logiciels utilisés

La partie réalisée sur MATLAB a été effectuée en salle informatique de Luminy sur un ordinateur travaillant sous Linux. Le package a donc été créé sous Linux et retravaillé dans un second temps sous Windows 8 sur un ordinateur personnel Asus.

Bien que le package *WATERStools_0.0* puisse être testé sous toutes les plateformes citées plus haut, le logiciel MATLAB reste indispensable pour faire tourner le programme. Les versions testées de MATLAB vont de R2006a à R2011b. MATLAB est un langage de programmation de quatrième génération et un environnement de développement ; il est utilisé à des fins de calcul numérique et permet de manipuler des matrices, d'afficher des courbes et des données, de mettre en œuvre des algorithmes, de créer des interfaces utilisateurs, et peut s'interfacer avec d'autres langages comme le C, C++, Java, et Fortran. Les commandes régissant le programme WATERS sont regroupées dans un fichier texte prenant la forme d'un script. Une des fonctionnalités principales de MATLAB utilisée fut l'option 'toolbox' qui permet comme son nom l'indique la création de boîte à outils. *WATERStools_0.0* en fait donc aujourd'hui partie et inclus en plus différents packages nécessaires au fonctionnement du logiciel comme *ROMSTOOLS_v3.1_02_2014*, *M_PACK* ou la librairie *Wavelab850*. En plus de ces packages, l'utilisation de l'outil 'm_map' a permis entre autre de réaliser des cartes réalistes et d'autres types de graphiques.

La partie réalisée sur LaTeX a été effectuée alternativement en salle informatique et sur un ordinateur personnel. Ce logiciel est un langage et un système de composition de document. Du fait de sa relative 'simplicité' il est très utilisé pour la rédaction de documents scientifiques, tel que le Userguide du logiciel WATERS. Le logiciel LaTeX possède son propre système d'encodage.

Une autre fonctionnalité intéressante de LaTeX est l'appel d'un outil de gestion de références bibliographiques servant à gérer et traiter des bases bibliographiques : BibTeX. Les références sont contenues dans un fichier d'extension *.bib et sont différentes selon les entrées. Ici, les entrées étaient généralement de type 'article'. L'outil BibTeX permet donc un référençage automatique et limite ainsi les erreurs dues à une réalisation à la main.

Fonctionnement de WATERS

Le fonctionnement du logiciel est tripartite :

- *L'analyse par ondelettes* (Wavelets analysis) : elle est utilisée afin de décomposer des cartes horizontales successives de la vorticité relative et d'en extraire des structures définies et localisées dans l'espace. L'algorithme utilisé fonctionne de la façon suivante : la meilleure base est choisie afin de minimiser le coût de la fonction (ici l'entropie de Shannon : c'est une fonction mathématique qui, intuitivement, correspond à la quantité d'information contenue ou délivrée par une source d'information), cela permet entre autre de trouver la meilleure localisation pour le paquet d'ondes. Ensuite, le modèle de vorticité relative est étendu dans cette base. Les ondelettes peuvent donc être triées comme des fonctions selon leur facteur d'ondes. Rappelons que seules les ondelettes ayant le plus gros facteur sont gardées en mémoire. Enfin, les structures sont extraites les unes après les autres en cherchant dans un premier temps le maximum du module de vorticité, et dans un second temps en diffusant la structure dans chaque direction de l'espace jusqu'à trouver un minimum de vorticité.

- *Le suivi du temps* (Time tracking) : le suivi des structures identifiées dans le temps requiert que le critère suivant soit satisfait :

$$\mathbf{c_{t,z=-200} \in \varepsilon_{t-\Delta t,z=-200}} \quad (1)$$

  où $c_{t,z=-200} \equiv (ic,jc)_{t,z=-200}$ représente le centre du tourbillon au temps t et à la profondeur -200m et $\varepsilon_{t-\Delta t}$ l'ensemble des points de la grille du même tourbillon à l'intervalle de temps précédent. La vitesse de translation du tourbillon est calculée comme étant la distance couverte par le centre du tourbillon dans Δt (deux jours d'intervalles). A chaque intervalle de temps de l'analyse, la vitesse de translation

instantanée *v* et le diamètre *D* du tourbillon sont enregistrées. Le critère (1) peut facilement être adapté pour le suivi dans le temps inverse, c'est-à-dire qu'un tourbillon identifié à un instant t peut être suivi en même temps avant et après cet instant t jusqu'à ce que le critère (1) ne soit plus vérifié.

- *Le suivi vertical* (Vertical Tracking) : L'analyse précédente est répétée à plusieurs profondeurs afin de diagnostiquer l'extension verticale d'un tourbillon identifié en 1er lieu à -200m. L'analyse commence à cette profondeur pour éviter que le modèle soit affecté par les interactions dynamiques qu'il peut exister entre l'atmosphère et la mer. Une structure identifiée au niveau z appartient à un tourbillon identifié auparavant au niveau z-Δz à condition que le critère suivant soit vérifié :

(2)

$$\mathbf{c_z \in \varepsilon_{z-\Delta z}}$$

où $c_z \equiv (ic,jc)_z$ représente le centre de la structure identifiée au niveau z et $\varepsilon_{z-\Delta z}$ l'ensemble des points de la grille du tourbillon sélectionné au niveau z-Δz. L'intervalle entre deux tranches horizontales est Δz=100m. Le suivi vertical s'arrête lorsque le signal de la vorticité relative du tourbillon devient trop faible (iz_L). Si ce signal ne faiblit pas, dans le but d'éviter une extension verticale excessive du tourbillon analysé, la profondeur maximale est fixée à z_{max}= -1000m. D'une manière générale, à un pas de temps fixé, l'aire A_k et le diamètre D_k sont tous les deux calculés à chaque niveau *k*, et le volume *V* du tourbillon peut être calculé de la manière suivante :

(3)

$$V = \sum_{k=1}^{izL} Ak \cdot \Delta z$$

La connaissance de la vitesse instantanée *v*, du diamètre *D*, et du volume *V* de chaque structure permet de calculer le transport instantané et le moment instantané.

Les données

Pour ce logiciel, les données d'entrée sont les champs de vitesse 3D produits par exemple par un modèle de circulation océanique. Nous avons utilisé des données de simulations obtenues à partir de l'outil Roms_agrif. Ces données correspondent à la configuration Benguela du Sud à haute résolution.

Résultats

Dans cette partie est relatée toutes les étapes qui ont servi à l'élaboration du package *WATERStools_0.0* ainsi que de sa mise à disposition pour tous les utilisateurs. De plus, afin que chaque utilisateur puisse se servir du package facilement, deux documents ont été réalisés lors du stage : l'un nommé 'readme.txt' expliquant les conditions générales d'utilisation (*cf* Annexes), l'autre 'Userguide.pdf' expliquant en détail la marche à suivre pour faire fonctionner le logiciel. Ce dernier fait partie des fichiers en téléchargement sur Internet à l'adresse http://www.com.univ-mrs.fr/~doglioli/.

Mise à jour du package

La première étape de la mise à jour fut la prise en main et la compréhension de tous les scripts. Ce travail fut le plus fastidieux car il a fallu lire des centaines de lignes de codes étant écrites par plusieurs personnes. La première mise à jour fut le changement des sorties graphiques. En effet, ces scripts ayant été rédigés il y a plusieurs années sur une version plus ancienne de MATLAB, les commandes devaient être modifiées ainsi que les "chemins" d'appels des différents fichiers servant à exécuter les figures. Chaque modification pouvait entraîner un disfonctionnement du logiciel, il a donc fallu avancer par tâtonnement, recompiler le logiciel et vérifier qu'il tournait toujours normalement.

De la même façon la dimension de l'aire de recherche devait être adaptée afin de fournir une modélisation plus claire.

Ce travail fut long et rigoureux car chaque script en appelant un autre, il fallait donc comprendre lesquels étaient appelés et lesquels ne l'étaient pas. Une autre modification importante fut l'ajout de certains package disponibles sur Internet mais non inclus dans le précédent package. En effet, afin de faire tourner le logiciel, certaines fonctions et librairies devaient être téléchargées séparément. Nous avons donc ajouté dans l'arborescence les packages *M_PACK*, *ROMSTOOLS_3.1_02_2014* et *Wavelab850*. En téléchargeant le paquet *WATERStools_0.0* on obtient l'arborescence suivante :

```
WATERS_data_example
WATERStools_0.0
    |- Wavelab850
    |- M_PACK
    |- ROMSTOOLS_v3.1_03_02_2014
    |- Documentation
        |- Userguide
    |-Figures
    |-fichiers.m
```

Explications :

- *WATERS_data_example* : inclus des sorties de fichiers obtenues à partir d'une simulation ROMS_Agrif (site internet : http://www.romsagrif.org); l'utilisation de cette installation correspond à la configuration Benguala du Sud à haute résolution;
- *M_PACK* est un package de MATLAB utilisé pour l'analyse d'harmoniques en série. De nombreuses fonctions appellent des données de NetCDF. Il a été inclus pour une visualisation en 3D des tourbillons identifiés par l'analyse en ondelettes (site internet : http://www.odyle.net/mma/archive/m-pack);
- *ROMSTOOLS* est une série d'outils utilisant un système de modélisation d'une région océanique (ROMS) issue de http://romasgrif.org. De plus, ce répertoire contient le package m_map qui correspond à un ensemble d'outils servant à la cartographie (http://www.eos.ubc.ca/~rich/map.html);
- *Wavelab850* est une collection de fonctions MATLAB utilisées pour créer une variété d'algorithmes liée à l'analyse en ondelettes (site internet : http://statweb.stanford.edu/~wavelab);
- *Documentation* : contient le guide d'utilisation du logiciel WATERS rédigé pendant ce stage, ainsi que son fichier source LaTeX ;
- *Figures* : répertoires par défaut des graphiques ;
- *fichiers.m* : fichiers scripts.

Mise à jour de la démo

Le principe de la démo de ce logiciel est de simuler le suivi d'un tourbillon océanique et d'enregistrer à chaque étape une figure le modélisant. La première idée a été d'ordonner l'affichage à chaque étape de la figure en 3D de la structure analysée.

Ci-après les figures apparaissant :

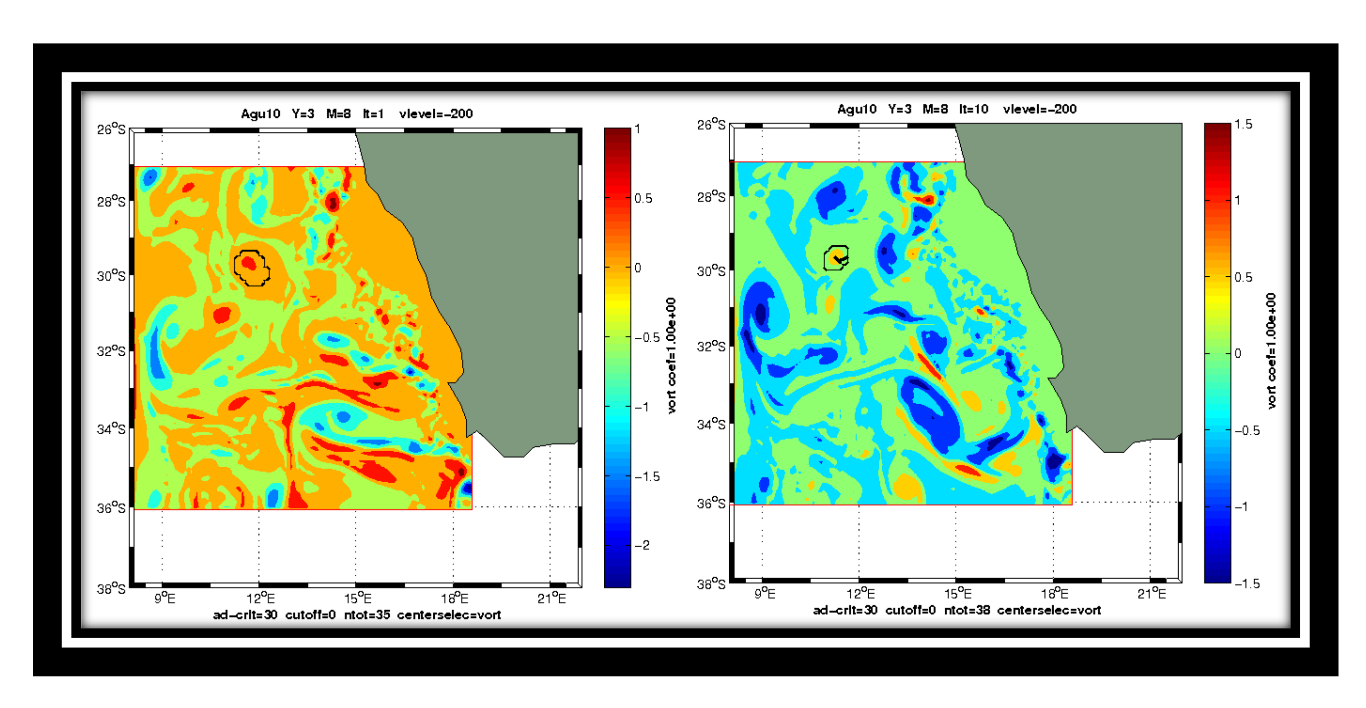


Figure 1 : carte de vorticité, contour du tourbillon en noir au début (à gauche) et après 10 itérations (à droite). La ligne noire (à droite) représente le déplacement de la structure.

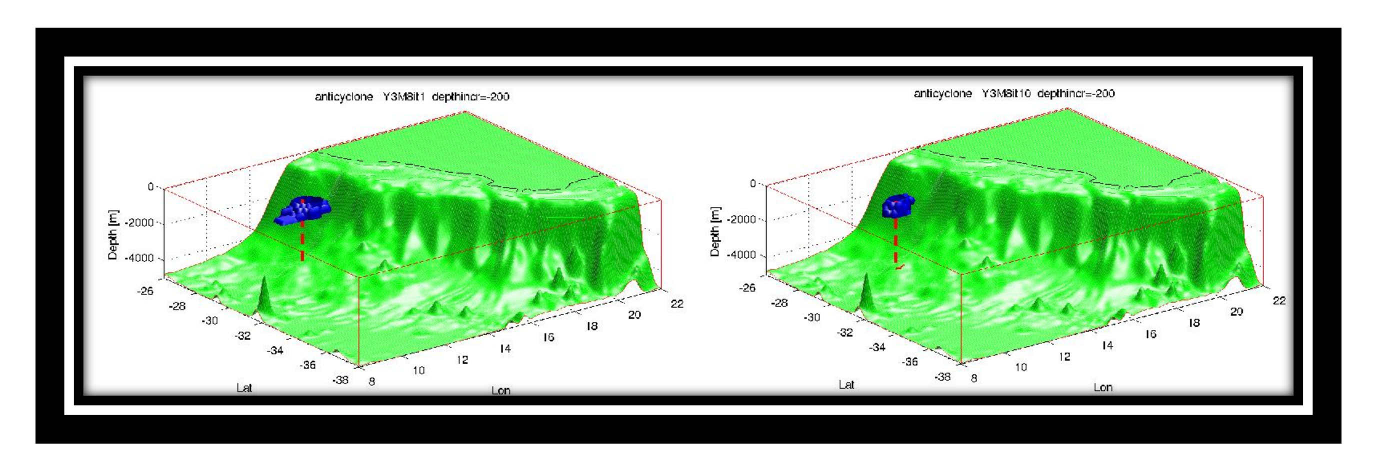


Figure 2 : première et dernière figure en 3D du tourbillon suivi

De la même façon, il existe un fichier nommé 'parameters3d.m' incluant tous les paramètres de bases pour le lancement du programme, comme la date d'enregistrement des données, l'aire de recherche, le nombre d'itérations à faire, etc. Nous avons modifié tous ces paramètres car les données n'étaient plus les mêmes.

Afin de rendre le programme plus interactif, nous avons aussi ajouté un outil de dialogue qui demande à l'utilisateur à chaque itération s'il souhaite continuer le suivi du tourbillon ou l'arrêter. En effet si celui-ci ne pense pas à changer les paramètres et donc le nombre d'itérations il lui suffit de cliquer sur 'Non'.

```
choice = questdlg('Do you want to continue the tracking ?','Alert','Yes','No','No');
    % Handle response
    if strcmp(choice,'No')==1;

        break

    end
```

Figure 3 : commandes affichant la boîte de dialogue citée plus haut

Une fois ces étapes réglées, le travail a consisté à tester cette démo et éliminer les petites erreurs qui persistaient. Par exemple, lorsque le tourbillon à suivre n'avait pas une vorticité suffisamment importante, il fallait indiquer au programme de ne plus ouvrir la boîte de dialogue ainsi que d'afficher tout de même la dernière figure 3D. Il suffisait alors de trouver les bonnes boucles dans les scripts.

Installation du package

Ce programme a été conçu pour MATLAB et nécessite au moins 32Mbites d'espace disque. Il a été testé sur des versions de MATLAB allant de R2006a à R2011b mais il peut être opérationnel sur des versions antérieures et plus récentes. Il fonctionne sous Linux, Windows et Mac. Pour éviter des problèmes de fonctionnement les package *'m_map'*, *ROMSTOOLS et Wavelab850* nécessaires ont été inclus dans *WATERStools_0.0* afin que chaque utilisateur ne soit pas obligé de le télécharger sur internet si il ne le possède pas déjà. Ainsi l'utilisation est simplifiée et directe. Pour installer le paquet il suffit de le télécharger librement à partir du site internet du projet WATERS. Un dossier compressé est téléchargé. La décompression fait apparaître le dossier *WATERStools_0.0* qui contient tous les programmes, données et figures du package. Pour l'utiliser, il faut ouvrir une session MATLAB et se placer dans le répertoire contenant le dossier *WATERStools_0.0* : Current folder : _*/WATERStools_0.0. Le logiciel est fin prêt à être utilisé. Un 'Usersguide.pdf' ('Guide d'utilisation') est mis à disposition dans le package si l'utilisateur rencontre un quelconque problème. De plus MATLAB contient une grande section d'aide appelé 'Help' pouvant aider chaque utilisateur.

Faire tourner la démo : tracking3d_ini.m

Le projet WATERS a été adapté pour lancer une démo. Il suffit de lancer le fichier 'tracking3d_ini' pour la démarrer.

Avant de lancer la démo, il est nécessaire d'ajouter de nouveaux chemins à partir de MATLAB. L'utilisateur doit choisir dans le navigateur de MATLAB 'new path', 'add with subfolders' et sélectionner dans l'arborescence 1) *ROMSTOOLS_v3.1_03_02_2014*, 2) *Wavelab850*, 3) dans le dossier *M_PACK*, sélectionner '*mfiles* 'et ajouter les sous-répertoires '*roms'* et '*toolsnc*'.

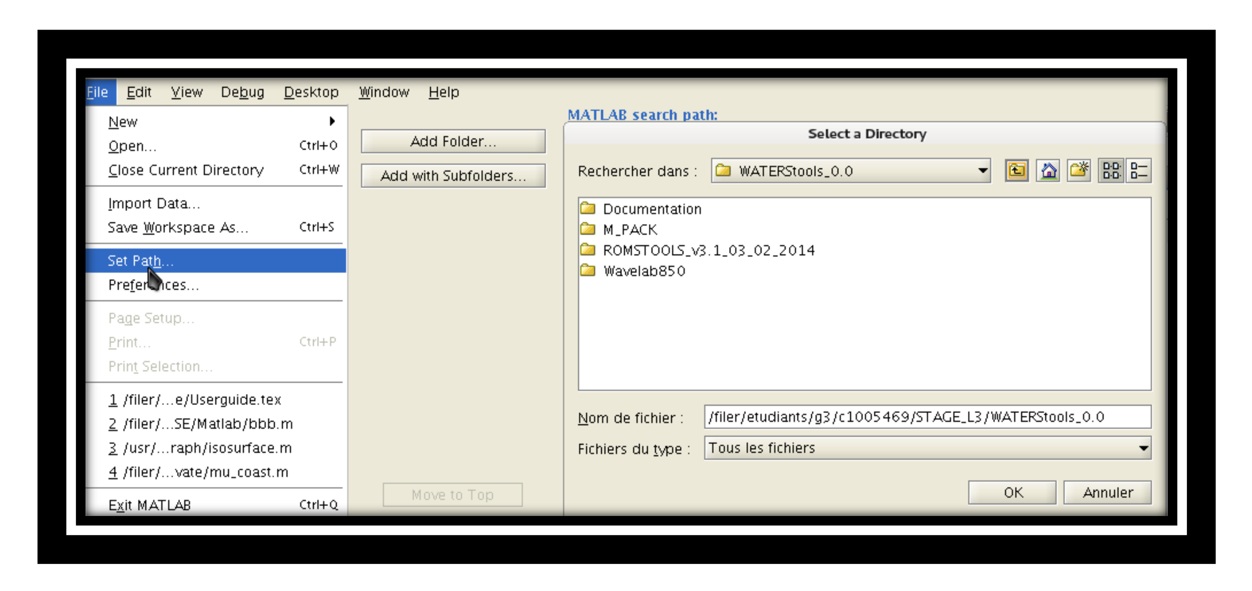


Figure 4 : écran de navigation de MATLAB avec les nouveaux chemins à ajouter

Quand la démo commence l'utilisateur appuis sur 'Entrée'. Les vecteurs de courant de vorticité et la côte d'une petite portion du Cape Basin devraient être représentés.

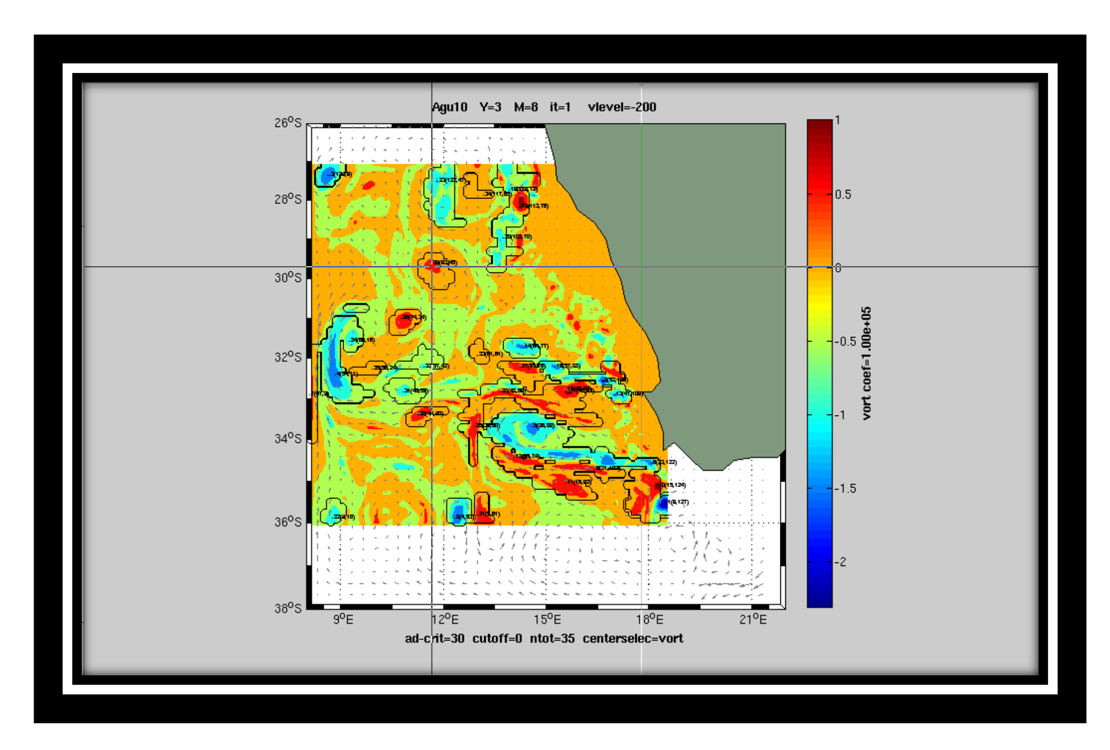


Figure 5 : première image de la démo : vecteurs de courant de vorticité. Les lignes noires représentent la périphérie de chaque structure identifiée par l'analyse en ondelettes. Dans nitre simulation, le tourbillon choisi est sous le pointeur.

L'utilisateur choisit ensuite le tourbillon qu'il souhaite suivre dans le temps et l'espace en cliquant avec le bouton droit de la souris sur la carte. Le clic gauche de la souris l'informe sur la localisation, le centre et le signe du tourbillon. La boîte de dialogue citée plus haut s'ouvre alors et demande à chaque étape de l'analyse si l'utilisateur souhaite continuer la démarche. Toutes les figures MATLAB (une pour chaque spécifications et pas de temps) sont enregistrées par défaut dans le répertoire *WATERtools_0.0*. La démo prend entre 3 et 4 minutes suivant la rapidité de l'utilisateur à répondre à la boîte de dialogue et la performance de l'ordinateur.

Discussion

La dernière modification officielle de ce paquet a été réalisée en 2007, soit il y a 7 ans. L'avantage de ce paquet est que tous les utilisateurs ont accès aux codes de chaque script. Ainsi chacun peut ajouter, enlever ou modifier un morceau de script afin qu'il colle le plus possible à l'utilisation que la personne veut en faire. En effet la démo a été faite avec une carte prédéfinie. Cependant si l'utilisateur souhaite tester cette démo avec une carte différente et qu'il possède les informations nécessaires à la réalisation de cette carte, il peut le faire. Si une autre personne change un calcul car il ne le trouve pas assez précis, il peut ensuite partager le logiciel contenant des modifications sur internet. Ce logiciel est donc un logiciel libre et c'est aussi ce qui fait son utilité auprès de chercheur du monde entier.

Le deuxième atout de ce logiciel est qu'il a été réalisé le plus simplement possible avec un guide d'utilisation en anglais inclus. Ainsi même quelqu'un qui n'est pas expert en physique océanographique peut le prendre en main facilement suite à la découverte du logiciel grâce à la démo.

Conclusion

Ce stage s'est terminé avec la première parution sur Internet de la démo qui est aujourd'hui disponible en téléchargement libre. Le travail effectué fut rigoureux et précis car les scripts et le 'Userguide' doivent être sans fautes et compréhensibles par tous. Il a donc fallu noter chaque petits changements ou modifications ce qui demande une grande rigueur. D'un point de vue informatique les progrès acquis pendant ce stage ont été notables de part une meilleur compréhension globale du logiciel MATLAB ainsi que du logiciel LaTeX. Il aura aussi fallu beaucoup réfléchir et se débrouiller par soi même pour trouver des solutions à des problèmes informatiques. En effet l'application d'un problème physique sur un outil informatique n'est pas toujours simple car parfois même si le calcul est théoriquement correct le logiciel ne l'applique pas comme l'utilisateur l'aurait voulu. Le projet de développer une telle boîte à outils n'avait jamais été réalisé auparavant il a donc fallu avancer parfois à tâtons en essayant beaucoup de choses, parfois sans succès mais c'est aussi ce qui fait avancer les choses. Il faut aussi être très pointilleux dans le travail pour ne laisser aucune erreur se glisser un code déjà très long.

Annexes

Fichier Readme.txt

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%                      WATERStools                      %%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%  This  file  is  free software ;  it  is distributed  in  %
%  the hope that  it  will  be  useful,  but  without  any  %
%  warranty.  You  can  redistribute  it and/or modify  it  %
%  under  the  terms  of  the  GNU  General Public License  %
%  as  published  by  the   Free  Software  Foundation  at  %
%       http://www.gnu.org/copyleft/gpl.html                %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
% The WATERS (Wavelet Analysis for Time-tracking Eddies in
% Regional modelS, 2004-2006) project has been supported by
% the programs LEFE/IDAO and LEFE/CYBER of the CNRS/INSU and
% by the Region PACA, France.
%
%        http://www.com.univ-mrs.fr/LOPB/WATERS
%
% 08/02/13: andrea.doglioli@univ-amu.fr: 1st official release
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Table of Contents
---------------------------------

   1. System Requirements

   2. Installation

   3. Help

   4. Tutorial: tracking3d_ini.m

   4. Contact Information


1. System Requirements
----------------------------------

This toolbox has been designed for Matlab. It needs at least 32 Mbites of
disk space.
It has been tested for Matlab versions ranging from R2006a to R2007a.
It has been mostly used on Linux workstations, but also used on Windows
computers.
The necessary Matlab toolbox m_map (vs 1.4, http://www.eos.ubc.ca/~rich/
map.html), other free programs and dataset from LATEX project are included.


2. Installation
--------------------------------
A simple uncompressing and untaring file (gunzip and tar -xvf) command is
useful to extract the file.
To use this software, open a Matlab session and make sure to be placed in
the right directory : Current folder : WATERStools_0.0 .
The software is ready to be used.
```

3. Tutorial: tracking3d_ini.m

Before running the demo, you have to set new path from Matlab. Choose 'add with subfolders' and select in the directory tree, i) {ROMSTOOLS_v3.1_03_02_2014}, ii) {Wavelab850}, iii) in the folder {M_PACK}, select mfiles and add the subfloders roms and toolsnc.
When the demo starts, press enter. The current vorticity vectors and the coastline of a small portion of the Cape Basin should be represented.
You can choose the eddy to tracking by right clicking with the mouse on the map. The left click informs you about the localization, the center and the sign of the eddy.
A dialog box asks you at each step of the analysis if you want to continue the eddy's tracking. If you want to stop it, click "No".

In the following, MATLAB figures will save in the {WATERtools_0.0} folder. One figure is recorded for each interval and for each specifications. MATLAB figures results show different periods in the life of the eddy. When the eddy moves over the continental slope, its translational velocity is weak, and its shape is more or less circular. The successive growth and reduction of its volume as well as a high varability of its diameter indicate that it incorporates and loses some mass.

The demo takes about 1 minute, depending on your rapidity on aswer to dialog boxes and on the performances of your computer.

4. Getting Help

If you are encountering any troubles to run programs, you can find more help in the LATEXtools Users Guide.
It is available in the folder WATERStools_0.0/Documentation directory.
Problems might also come from your different version of Matlab. Matlab includes an extensive online help system. Use the Help menu in the Matlab menu bar to access Help.

5. Contact Information

For contact information visits :
http://www.com.univ-mrs.fr/~doglioli/

Fichier parameters3d.m

```
%%%% Andrea %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% 13/07/07 %%%%%%%%
%%%%           EDDY TRACKING 3D PACKAGE                    %%%
%%%%                                vs 3D.1.1              %%%
%%%% parameters3d                                          %%%
%%%% tracking3d_ini                                        %%%
%%%%  |_anim_tracking3d<-ad_extracteur<-census_wlt         %%%
%%%%     |_tracking3d                                      %%%
%%%%     |_eddysection3d                                   %%%
%%%%     |_trackingvert                                    %%%
%%%%                                                       %%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Reference:                                                %
%% Doglioli A.M., Blanke B., Speich S., Lapeyre G. (2007), %
%% Tracking coherent structures in a regional ocean model  %
%% with wavelet analysis: application to Cape Basin Eddies.%
%% J.Geophys.Res., 112, C05043, doi:10.1029/2006JC003952.  %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% This  file  is  free software ;  it  is distributed  in %
%% the hope that  it  will  be  useful,  but  without  any %
%% warranty.  You  can  redistribute  it and/or modify  it %
%% under  the  terms  of  the  GNU  General Public License %
%% as  published  by  the   Free  Software  Foundation  at %
%%       http://www.gnu.org/copyleft/gpl.html.             %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

clear;close all;
%
%--- source files
simu='Agu10'; %%% conventional name
gridfile='/filer/etudiants/g3/c1005469/STAGE_L3/WATERS_data_example/roms_grd_VH
R.nc';
datafile='/filer/etudiants/g3/c1005469/STAGE_L3/WATERS_data_example/roms_avg_VH
R';

%
%--- time parameters
cycledir='fwd'; %%% temporal direction of analisys -> bkw or fwd
fr0=0;          %%% number of first frame !!! fr-1
Ymin=3;         %%% first year
Mmin=8;         %%% first month
%Dmin=30;       %%% only for GASC_POP25 output file
itmin=1;        %%% first interval
Ymax=3;         %%% last year
Mmax=8;         %%% last month
%Dmax=30;       %%% only for GASC_POP25 output file
itmax=10;       %%% last interval
vlevel=-200;    %%% shallowest vertical level %%% [m]
%
%--- 3D parameters
Zmin=-1001;     %%% deepest vertical level %%% [m] DA TOGLIERE
depthincr=-200; %%% distance between two successive horizontal slices %%% [m]
%
%--- analysis setup
extraction='old'; %%% kind of wavelet exraction %%% [new/old] WARNING new is
accurate but time consuming
factor=0.25;      %%% of number of spectral coefs
varname='vort';   %%% wavelets analysis variable ->
vorp,vori,vort(BEST),salt,temp,zeta NB:vori only forward
coef=10^5;        %%% var rescaling
interval=[-2.5:0.5:2.5];%%% contour plot
delta=2^7;        %%% dim of area of research %%% [grid meshes]
jmin=30;          %%% grid coordinates of bottom left corner of analysis area
imin=1;         %%%  "       "       "    "     "     "    "     "      "
```

```
%
centerselec='vort';%%% center definition: max(abs(vorticity)) or min(speed) %%%
[vort/spee]
ad_crit=30;         %%% min surface of an eddy %%% [grid meshes]
cutoff=0.;          %%% var cutoff %%% [*coef]
%
%--- plot
varname2='vort';    %%% contour plot variable %%%
[vorp,vori,vort,salt,temp,zeta]
coef2=1;            %%% var2 rescaling
interval2=[35.3:0.02:36.3];%%% contour levels %%% [units of var2]
contouring='y';     %%% contour around eddy %%% [y/n]
cross='n';          %%% plot a cross as reference of vertical sections %%% [y/n]
sect=1;             %%% vertical section extention %%% [lat& lon degree]
numbering='n';      %%% put the eddy number in eddy meshes %%% [y/n]  |
freqnumbering=3;    %%% every freqnumbering meshes                   <-
%
lonrange1=[8 22];    %%% lon range of first figure
latrange1=[-38 -26];%%% lat   "   "    "     "
lonrange2=[8 22];    %%% lon range of other figures
latrange2=[-38 -26];%%% lat   "   "    "     "
%
%--- output file names
moviename='animation';
sectionsname='sections';
%
%%%%%%%%% DON'T TOUCH BELOW %%%%%%%%%%%%%%%%%%
jmax=jmin+delta-1;
imax=imin+delta-1;
%
nc=netcdf(gridfile);
topo=nc{'h'}(:);
topo=-topo;
pm=nc{'pm'}(:);
pn=nc{'pn'}(:);
angle=nc{'angle'}(:);
close(nc)
%
anomalycalc=0;
periodicity=0;
filter_vort=1;
fierl='si';
%--- for Ariane
lmt=99999;          %ariane initial positions in vertical tracking
timedir='BW';    % "
%
navgdays=2; %[days] print frequency
skipanim=1;
rempts=[0 0 0 0];
%
scrsz = get(0,'ScreenSize');
figure(1);set(1,'Visible','on','Position',[scrsz(3)/3 scrsz(4) scrsz(3)/1.5
scrsz(4)/1.5])
figure(1);hold on;
m_proj('mercator','lon',lonrange1,'lat',latrange1) %all Agu10
%m_grid('color',[0 0 0]);
m_grid('box','fancy');
%--- coastline
%m_gshhs_c('patch',[.5 .6 .5]);
%m_gshhs_h('patch',[.5 .6 .5]);
%ad_add_towns('town.dat',6,min(min(lon)),19,min(min(lat)),max(max(lat)))

save parameters3d.mat
```

Références

Doglioli, A. M., Blanke, B., Speich, S., and Lapeyre, G. (2007). Tracking coherent structures in a regional ocean model with wavelet analysis: application to Cape Basin Eddies. J. Geophys. Res., **112**, C05043, doi: 10.1029/2006JC003952.

Guilherme P. Castelão, Luiz C. Irber Jr., A. B. M. V. B. (2013). An objective reference system for studying rings in the ocean. Computers & Geosciences, 61(0):43-49.

Leeuw, W. C. and Post, F. H.: A statistical view on vectors fields. Visualization in scientific Computing Springer-Verlag, 1995.

McWilliams, J. C., 1990: The vortices of two-dimensional turbulence. *J. Fluid Mech*., **219**, 361-385.

Nencioli, F. Dong, C. D. T. W. L. M. J. C. (2010). A vector geometry-based eddy detection algorithm and its application to a high-resolution numerical model product and high-frequency radar surface velocities in the southern california bight. J. Atmos. Ocean. Technol., 27:564-579.

Penven, P., V. Echevin, J. Pasapera, F. Colas, and J. Tam (2005), Average circulation, seasonal cycle, and mesoscale dynamics of the Peru Current System: A modeling approach, *J. Geophys. Res*., *110*, C10021, doi : 10.1029/2005JC002945.

Sadarjoen, I. A. and Post, F. H.: detection, quantification, and tracking of vortices using streamline geometry, Comput. Graph., 24, 333-341, 2000.

Siegel, A. and Weiss, J. B. (1997). A wavelet-packet census algorithm for calcu- lating vortex statistics. Physics of Fluids, 9:1988-1999.

Souza, J. M. A. C., de Boyer Montégut, C., and Le Traon, P. Y. (2011). Com- parison between three implementations of automatic identication algorithms for the quantication and characterization of mesoscale eddies in the south atlantic ocean. Ocean Science, 7(3):317-334.